

Claims

WHAT IS CLAIMED IS:

1 1. A method for filtering one or more messages for transmission to a subscriber
2 computing system according to an individual information request criteria, the method comprising:
3 constructing a binary decision diagram implication graph for each individual information
4 request criteria specified for each subscriber;
5 identifying logical implications from one or more nodes in a binary decision diagram from a
6 first information request criteria to one or more corresponding binary decision diagrams within a
7 second information request criteria;
8 receiving one or more messages to be filtered;
9 evaluating a first information request criteria based upon information within the received
10 messages;
11 evaluating one or more information request criteria based upon information within the
12 received messages using the identified logical implications between one or more binary decision
13 diagrams within the information request criteria being evaluated and one or more binary decision
14 diagrams previously evaluated; and
15 transmitting the received message to the subscriber computing system corresponding to an
16 information request criteria evaluated to be satisfied by information contained within the received
17 message.

1 2. The method according to claim 1, wherein the binary decision diagrams include an
2 expression of an information request criteria in an if-then-else normal form.

1 3. The method according to claim 2, wherein the constructing step further comprises:
2 recursively visiting the high and low successors for each node in the binary decision
3 diagrams;
4 while visiting each node, determine the precondition $\text{pre}(X')$ for each successor and compute
5 the target $t(X')$ for all visited nodes and apply permissible implications; and
6 iterate the processing for all implications.

1 4. The method according to claim 3, wherein permissible implications for a node M with
2 successor node N include:
3 if node N is equal to the high successor $\text{high}(M)$, and
4 if $p(M)$ implies $p(N)$, then remove N and set the $\text{high}(M)$ equal to $\text{high}(N)$; and
5 if $p(M)$ implies $\neg p(N)$, then remove N and set $\text{high}(M)$ equal to $\text{low}(N)$.

1 5. The method according to claim 3, wherein permissible implications for a node M with
2 successor node N include:
3 if node N is equal to the low successor $\text{low}(M)$, and
4 if $\neg p(M)$ implies $p(N)$, then remove N and set the $\text{low}(M)$ equal to $\text{high}(N)$; and
5 if $\neg p(M)$ implies $\neg p(N)$, then remove N and set $\text{low}(M)$ equal to $\text{low}(N)$.

1 6. The method according to claim 5, wherein the evaluating steps further comprises:
2 determining if a current node is a leaf node in the binary decision diagram;

3 if the current node is a leaf node, marking the information request criteria as being decided
4 and returning the value of the current node;
5 if the current node is not a leaf node, determining a value of the expression for the node
6 $p(X)$;
7 if the value of the expression of the node is true
8 setting $X' = \text{high}(X)$ otherwise $X' = \text{low}(X)$;
9 inserting X' into the rank; and
10 visiting the targets of node X' to compare the current node with the target node;
11 if the target node is lower than the current node according to a predicate order, update
12 the current node.

1 7. A computer program product readable by a computing system and encoding
2 instructions for filtering one or more messages to be transmitted to a subscriber computing system
3 according to an individual information request criteria, the computing process comprising:
4 constructing a binary decision diagram implication graph for each individual information
5 request criteria specified for each subscriber;
6 identifying logical implications from one or more nodes in a binary decision diagram from a
7 first information request criteria to one or more corresponding binary decision diagrams within a
8 second information request criteria;
9 receiving one or more messages to be filtered;
10 evaluating a first information request criteria based upon information within the received
11 messages;

evaluating one or more information request criteria based upon information within the received messages using the identified logical implications between one or more binary decision diagrams within the information request criteria being evaluated and one or more binary decision diagrams previously evaluated; and
transmitting the received message to the subscriber computing system corresponding to an information request criteria evaluated to be satisfied by information contained within the received message.

8. The computer program product according to claim 7, wherein the binary decision diagrams include an expression of an information request criteria in an if-then-else normal form.

9. The computer program product according to claim 8, wherein the constructing step further comprises:

recursively visiting the high and low successors for each node in the binary decision diagrams;

while visiting each node, determine the precondition $\text{pre}(X')$ for each successor and compute the target $t(X')$ for all visited nodes and apply permissible implications; and
iterate the processing for all implications.

10. The computer program product according to claim 9, wherein permissible implications for a node M with successor node N include:

if node N is equal to the high successor $\text{high}(M)$, and

if $p(M)$ implies $p(N)$, then remove N and set the $\text{high}(M)$ equal to $\text{high}(N)$; and

if $p(M)$ implies $\neg p(N)$, then remove N and set $\text{high}(M)$ equal to $\text{low}(N)$.

11. The computer program product according to claim 9, wherein permissible implications for a node M with successor node N include:

if node N is equal to the low successor low(M), and

if ! p(M) implies p(N), then remove N and set the low(M) equal to high(N); and

if ! p(M) implies ! p(N), then remove N and set low(M) equal to low(N).

12. The computer program product according to claim 9, wherein the evaluating steps further comprises:

determining if a current node is a leaf node in the binary decision diagram;

if the current node is a leaf node, marking the information request criteria as being decided and returning the value of the current node;

if the current node is not a leaf node, determining a value of the expression for the node p(X);

if the value of the expression of the node is true

setting $X' = \text{high}(X)$ otherwise $X' = \text{low}(X)$;

inserting X' into the rank; and

visiting the targets of node X' to compare the current node with the target node;

if the target node is lower than the current node according to a predicate order, update the current node.

13. A publication-subscription broker server computing system for filtering one or more messages to be transmitted to a subscriber computing system according to an individual information request criteria, the broker server computing system comprises:

4 a memory module;
5 a mass storage system; and
6 a programmable processing module, the programmable processing module performing a
7 sequence of operations to implement the following:
8 constructing a binary decision diagram implication graph for each individual
9 information request criteria specified for each subscriber;
10 identifying logical implications from one or more nodes in a binary decision diagram
11 from a first information request criteria to one or more corresponding binary decision
12 diagrams within a second information request criteria;
13 receiving one or more messages to be filtered;
14 evaluating a first information request criteria based upon information within the
15 received messages;
16 evaluating one or more information request criteria based upon information within the
17 received messages using the identified logical implications between one or more binary
18 decision diagrams within the information request criteria being evaluated and one or more
19 binary decision diagrams previously evaluated; and
20 transmitting the received message to the subscriber computing system corresponding
21 to an information request criteria evaluated to be satisfied by information contained within
22 the received message.

23 14. The broker server computing system according to claim 13, wherein the binary
24 decision diagrams include an expression of an information request criteria in an if-then-else normal
25 form.

1 15. The broker server computing system according to claim 13, wherein the constructing
2 the implication graph further comprises:

3 recursively visiting the high and low successors for each node in the binary decision
4 diagrams;

5 while visiting each node, determine the precondition $\text{pre}(X')$ for each successor and compute
6 the target $t(X')$ for all visited nodes and apply permissible implications; and

7 iterate the processing for all implications.

1 16. The broker server computing system according to claim 14, wherein permissible
2 implications for a node M with successor node N include:

3 if node N is equal to the high successor $\text{high}(M)$, and

4 if $p(M)$ implies $p(N)$, then remove N and set the $\text{high}(M)$ equal to $\text{high}(N)$; and

5 if $p(M)$ implies $\neg p(N)$, then remove N and set $\text{high}(M)$ equal to $\text{low}(N)$.

1 17. The broker server computing system according to claim 15, wherein permissible implications
2 for a node M with successor node N include:

3 if node N is equal to the low successor $\text{low}(M)$, and

4 if $\neg p(M)$ implies $p(N)$, then remove N and set the $\text{low}(M)$ equal to $\text{high}(N)$; and

5 if $\neg p(M)$ implies $\neg p(N)$, then remove N and set $\text{low}(M)$ equal to $\text{low}(N)$.

1 18. The broker server computing system according to claim 15, wherein the evaluating steps
2 further comprises:

3 determining if a current node is a leaf node in the binary decision diagram;

4 if the current node is a leaf node, marking the information request criteria as being decided
5 and returning the value of the current node;
6 if the current node is not a leaf node, determining a value of the expression for the node
7 $p(X)$;
8 if the value of the expression of the node is true
9 setting $X' = \text{high}(X)$ otherwise $X' = \text{low}(X)$;
10 inserting X' into the rank; and
11 visiting the targets of node X' to compare the current node with the target node;
12 if the target node is lower than the current node according to a predicate order, update
13 the current node.